

Estudio cualitativo de arquitecturas orientadas a computación móvil

Mónica Aydé Vallejo Velásquez

José Edinson Aedo Cobo

Grupo de Microelectrónica y Control, Universidad de Antioquia

Fecha de recepción: 04-08-2005

Fecha de aceptación: 05-12-2005

ABSTRACT

The increase of the wireless networks has created a significant growth in demand for portable devices with different types of services such as text processing, connectivity to different networks, remote data base access and multimedia processing. In order to respond to these requirements, a number of different hardware architectures have been proposed recently, designed principally to integrate useful characteristics such as high performance for multimedia functions, energy efficient handling, small size and a low complexity. This work presents a general analysis of several architectures designed principally for real time multimedia proces-

sing in mobile applications. This article describes the advantages of these architectures and raises comparison strategy among them.

KEY WORDS

Mobile Computing, real time multimedia processing, power management, multimedia applications.

RESUMEN

El incremento de las redes inalámbricas ha creado un significativo incremento en la demanda de dispositivos portátiles que integran diferentes tipos de servicios tales como procesamiento de texto, acceso a bases de datos remotas y procesamiento

multimedia. Para responder a estos requerimientos tecnológicos, han sido propuestas diferentes arquitecturas en los últimos años, diseñadas principalmente para integrar características útiles tales como alto desempeño para funciones multimedia, manejo eficiente de la energía, tamaño pequeño y baja complejidad. En este trabajo se presenta un análisis general de algunas arquitecturas especialmente diseñadas para el procesamiento multimedia en tiempo real,

para aplicaciones móviles. El artículo describe sus bondades y plantea una estrategia de comparación entre ellas.

PALABRAS CLAVE

Computación móvil, procesamiento multimedia en tiempo real, administración del consumo de potencia, aplicaciones multimedia.

Clasificación Colciencias: A

I. INTRODUCCIÓN

En el panorama actual, las crecientes exigencias de la sociedad de la información para acceder desde un ambiente móvil a diferentes servicios computacionales han motivado el desarrollo de sistemas convergentes que permitan soportar de forma integrada diversas aplicaciones como funciones multimedia y conectividad a redes inalámbricas. Para responder a estos requerimientos tecnológicos, distintas arquitecturas para sistemas embebidos han sido propuestas en la literatura en los últimos años, todas ellas orientadas a ofrecer un alto desempeño en el procesamiento de datos multimedia en tiempo real, manejo eficiente de la energía, tamaño pequeño y baja complejidad en el diseño.¹

En la actualidad se observa que el diseño de arquitecturas orientadas a aplicaciones móviles se caracteriza por integrar en un solo chip diferentes módulos de *hardware* (*System-on-Chip* SoC). Esta tendencia implica nuevos retos para los diseñadores, ya que resulta crítico no sólo el acoplamiento de módulos de procesamiento multimedia sino también el diseño de plataformas de comunicación *on-chip* óptimas, que permitan el manejo eficiente de un gran flujo de datos.

El rango de arquitecturas propuestas para aplicaciones móviles va desde procesadores de aplicación específica, procesadores de propósito general (GPP) con extensiones multimedia (conocidos también como GPP de instrucciones con múltiples datos (SIMD)),^{2,3} procesadores multimedia de propósito especial,⁴ procesadores superescalares, procesadores tipo VLIW (*Very Long Instruction Word*)⁵

y plataformas completamente reconfigurables. Sin embargo, el rango de arquitecturas que han demostrado ser óptimas para responder acertadamente a las restricciones impuestas por las aplicaciones móviles es más reducido que la clasificación presentada anteriormente. En el caso de las arquitecturas superescalares y VLIW tradicionales, generalmente trabajan a altas frecuencias de reloj y usan complejas estructuras para la implementación de paralelismo, lo cual se traduce en un alto consumo de energía y un incremento de los costos de la arquitectura.⁵

Para el caso de las arquitecturas tradicionales SIMD, el alto costo que puede involucrar el proceso de reorganización de datos necesario en algunos casos puede introducir grandes restricciones haciéndolos poco viables.²

Los procesadores multimedia de propósito especial tales como el STN8800 de STMicroelectronics,⁶ el S3C24A0 de Samsung,⁷ el Dragonball MC9328MX21 de Freescale,⁸ el PXA27x de Intel,⁹ usualmente tienen módulos de hardware en forma de periféricos para realizar diferentes funciones de procesamiento, lo que permite obtener un mejor desempeño.

Este artículo ha sido organizado de la siguiente forma: en la sección II se realiza una descripción general de las características principales de las aplicaciones multimedia. En la sección III se describen algunas de las arquitecturas más utilizadas actualmente para aplicaciones móviles y sus características más sobresalientes. En la sección IV se analizan las arquitecturas descritas anteriormente. Por

último, en la sección VI se presentan las conclusiones y opciones futuras en esta área de trabajo.

II. CARACTERÍSTICAS DE APLICACIONES MULTIMEDIA

En general las principales características de las aplicaciones multimedia son:

- Respuesta en tiempo real: estas aplicaciones exigen un alto desempeño en la realización de tareas de tiempo real tales como videoconferencia y comercio electrónico, por tal motivo es necesario garantizar el desempeño suficiente para ofrecer una buena calidad del servicio (QoS).^{2,10}
- Tipos de datos “continuous-media”: las funciones media procesan tipos de datos “continuous stream” generalmente de tamaño de 8 a 16 bits con localidad temporal limitada.¹
- Paralelismo de grano fino y grano grueso: requieren gran capacidad de procesamiento paralelo de datos con localidad temporal limitada y alta tasa de accesos a memoria.^{2,11}].
- Las aplicaciones multimedia son computacionalmente intensivas, ya que frecuentemente realizan de 100 a 200 operaciones aritméticas para cada dato leído desde memoria.¹² La Tabla 1 tomada de¹³ describe la distribución de operaciones necesarias en aplicaciones multimedia.
- Alta localidad de referencia de instrucciones: las funciones tipo *media* usualmente ejecutan diferentes procesos que se repiten con gran frecuencia en la aplicación,

esto demuestra una alta localidad temporal y espacial de instrucciones.¹⁴

- Ancho de banda de memoria: las funciones tipo *media* requieren procesar grandes conjuntos de datos.

Tabla 1. Distribución de operaciones en aplicaciones multimedia¹³

Tipo de operación	Frecuencia de operación
ALU	40%
Load/Store	26 -27 %
Branch	20%
Shift	10%
Multiplicación entera	3 - 4 %
Punto flotante	2%

III. ARQUITECTURAS PARA APLICACIONES MULTIMEDIA

Recientemente la literatura técnica y científica reporta un número creciente de procesadores y arquitecturas multimedia de propósito especial que han sido diseñados con fines académicos y comerciales. Las opciones son muchas, pero en este artículo hemos tomado una muestra de cuatro arquitecturas para realizar un análisis de las características más relevantes en cuanto al manejo de datos multimedia con el fin de obtener una aproximación apropiada en la búsqueda de una arquitectura mejorada para aplicaciones de computación móvil.

A. Arquitectura del acompañante digital móvil, proyecto Moby Dick

La arquitectura está formada por un procesador de propósito general y módulos reconfigurables autónomos (módulos de red, *display*, audio y video), todos ellos interconectados por un suiche denominado *Octopus*.¹⁵ Los módulos son propuestos para integrar modos de conservación de energía y esquemas dinámicos de adaptación al ambiente de conexión y a los requerimientos del usuario.

Específicamente esta arquitectura presenta un diseño centrado en la conexión,¹⁵ ya que implementa como

esquema de comunicación un suiche, el cual permite reducir el tráfico de datos mediante la eliminación de copias innecesarias al remover la CPU del flujo de datos, mas no del flujo de control. De esta forma, la CPU es reducida a operar como un controlador que inicializa el sistema, maneja la calidad del servicio (QoS) asociado con cada conexión y maneja la ejecución de algunos protocolos complejos.

Un prototipo parcial de baja complejidad de la arquitectura, consistió de un conjunto de pequeños sistemas que contienen FPGAs, microcontroladores y memorias. A continuación

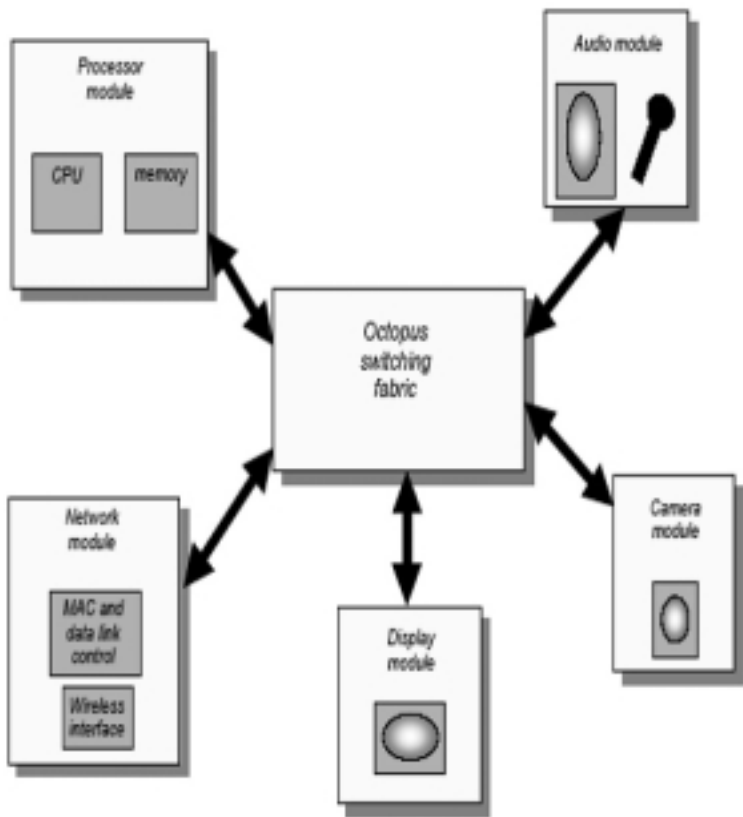


Figura 1. Arquitectura del acompañante digital móvil Moby Dick.¹⁵

se describen algunas características y requerimientos de estos subsistemas

El módulo de red incluye un Modem WaveLAN en la capa física, que proporciona la funcionalidad básica para enviar y recibir *frames* de datos en modo *half-duplex* a rata de 2 Mb/s; una memoria SRAM de 512 Kbytes que actúa como pila de conexión; una FPGA Xilinx XC4010 [16] que controla el flujo de datos entre el *Modem* y el *Host* y proporciona funciones de corrección de errores; un microcontrolador PIC 16C66 el cual implementa el control de tráfico, el control del enlace de datos, y el protocolo MAC eficiente en energía E²MaC, que permite la conexión del sistema a una estación base a través de una red inalámbrica ATM con un QoS apropiado.¹⁵

El módulo de *display* es de tamaño pequeño y baja resolución e incluye *hardware* de decodificación y posprocesamiento.

El suiche *Octopus* es un sistema reconfigurable que conforma una red interna de comunicación similar a un suiche ATM simplificado de 8 x 8, que garantiza un ancho de banda suficiente para aplicaciones multimedia. *Octopus* está formado por dos subsistemas: un *switching fabric* de 8 x 8 que es implementado mediante la FPGA XC4010XL de Xilinx¹⁶ y el cual se encarga de enrutar el tráfico entre los distintos módulos que forman el sistema; y ocho módulos controladores de interfaz (MIC) implementados con microcontroladores microchip PIC 16C66, conectados a los puertos del suiche, que realizan operaciones de establecimiento de conexión, arbitraje e interfaz con los diferentes subsistemas. En la estruc-

tura del suiche las conexiones internas son identificadas con una etiqueta que especifica el tipo de dato y la dirección del módulo destino, lo que permite eliminar la transferencia de un gran número de bits de direcciones por acceso.

En general, el esquema de diseño planteado para el suiche *Octopus* le permite soportar tres conexiones simultáneas entre módulos disjuntos con un rendimiento máximo de 96 Mb/s.

Como aspecto importante de esta propuesta cabe anotar que la implementación descrita anteriormente fue realizada únicamente con el objetivo de validar funcionalmente el prototipo. Por lo tanto, ésta es ineficiente desde el punto de vista del consumo de potencia, ya que las FPGA usadas no son adecuadas para sistemas móviles.

B. Procesador Nomadik STN8800 de STMicroelectronics

La familia de procesadores multimedia Nomadik está formada por cuatro procesadores: el STn8800, salió al mercado en el 2003 y los nuevos procesadores STn8810, STn8811 y STn8812 están siendo lanzados actualmente (2005).¹⁷

Los procesadores de STMicroelectronics disponen de una arquitectura abierta de procesamiento distribuido, que usa aceleradores programables inteligentes para la realización en forma paralela de funciones intensivas en datos y un procesador de propósito general para el control. Como módulo CPU, el STn8800, utiliza el *cores* ARM926E-JS de 32 bits de la familia ARM9 que incluye MMU, *cache* de instrucciones de 32KB, *cache* de datos de 16KB, multiplicador de

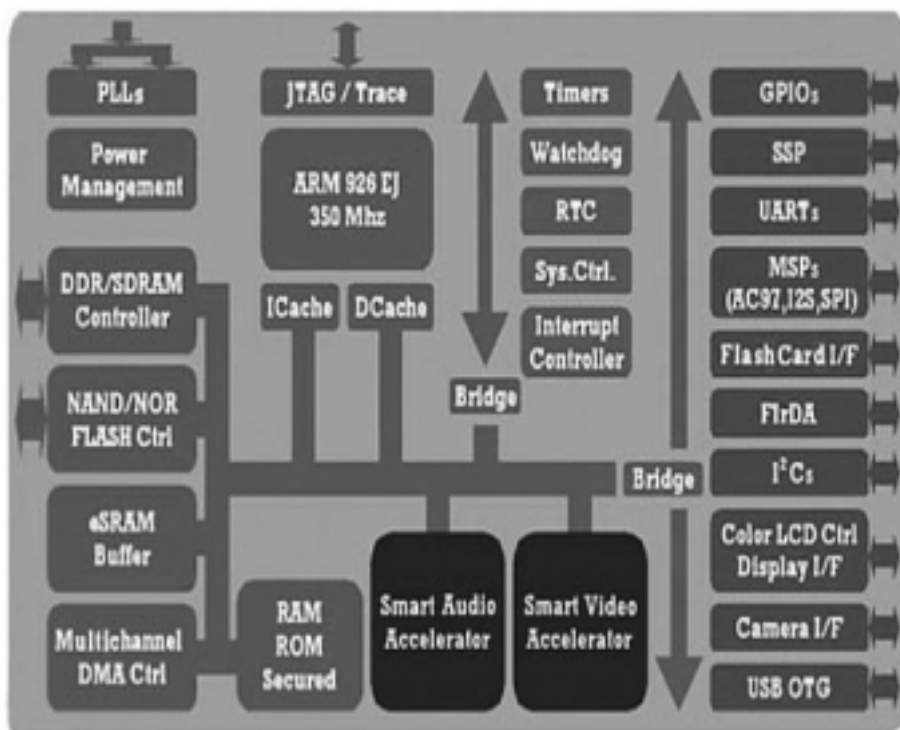


Figura 2. Arquitectura del procesador Nomadik.²¹

16 x 32 bits que realiza operaciones MAC en un ciclo y un acelerador Java.¹⁸ Adicionalmente la plataforma ARM integra una arquitectura de comunicación multicapa basada en bus conocida como AMBA (*Advanced Microcontroller Bus Architecture*) con un bus tipo *Advanced High-performance Bus* (AHB).^{19,20} Este subsistema consiste en un esquema de buses multicapa síncrono que permite realizar transferencias de datos entre módulos del sistema sin la intervención de la CPU en el flujo de datos.

Un sistema AMBA-AHB típico contiene los siguientes componentes: *AHB master*, inicializa el sistema para realizar las operaciones de lectura y escritura mediante la activación de unas señales de dirección y

control; *AHB slave*, responde a las operaciones solicitadas por el *AHB master*; *AHB arbiter*, garantiza que solamente un maestro tenga acceso al bus de comunicación en un mismo tiempo mediante esquemas de asignación de prioridades y algoritmos de arbitraje; *AHB decoder*, decodifica la dirección de cada transferencia y selecciona el dispositivo esclavo involucrado en la transferencia.

Específicamente, el STN8800, está basado en un AMBA-AHB de nueve capas, de las cuales dos son usadas por la CPU proporcionando rutas para acceder la memoria externa y otras dos le permiten acceder a periféricos de aplicación.

El procesador STN8800 posee dos módulos aceleradores, uno para audio

y otro para video. Estos están basados en el MMDSP+ *core*, el cual es un procesador DSP comercial de 75MHz tipo VLIW totalmente programable.²² Específicamente el acelerador de audio soporta entre su amplia librería de *software* los estándares MPEG3, MIDI, ACC y MPEG1. El acelerador de video es un híbrido *hardware/software* que incluyen tanto al MMDSP+ *core* como módulos de *hardware* dedicado que desempeñan funciones de pre-procesamiento, post-procesamiento, codificación y decodificación.

Los procesadores de STMicroelectronics incorporan diferentes técnicas de diseño orientadas a reducir el consumo de energía, entre ellas se destacan: el uso de memoria *on-chip* como buffer de *cache*; el uso de modos de operación de bajo consumo de la CPU; la implementación de la técnica de *clock gating* y la reducción del voltaje de alimentación.

C. Procesador PXA270 de Intel

La tecnología Intel ha respondido a los requerimientos de las aplicaciones multimedia mediante el diseño de las familias de procesadores PXA255^{23,24}, PXA26x²⁵ y PXA27x²⁶, los cuales poseen características especiales que les permiten soportar las exigencias del procesamiento multimedia en ambientes móviles.

Como se muestra en la Figura 3, específicamente la familia PXA27x, integra un *core XScale* de Intel como módulo CPU, una memoria SRAM *on-chip* de 256KB la cual es usada como *frame buffer* para aplicaciones gráficas y de video, diversos módulos multimedia (e.g. controlador de LCD e interfaz de cámara) y diferentes interfaces de periféricos (e.g. UART,

USB, AC97). Toda la arquitectura ha sido mejorada por un conjunto de nuevas tecnologías entre las cuales se incluyen: 1) la tecnología *Wireless MMX™*, que proporciona funcionalidades adicionales para aplicaciones como juegos 3D y video avanzado mientras optimiza el consumo de energía;²⁷ 2) la tecnología *Wireless SpeedStep*, que implementa técnicas avanzadas de reducción de consumo de potencia;^{28,29} 3) la tecnología *Quick Capture*, que proporciona un incremento en las capacidades y la calidad de las cámaras que soporta el procesador;^{30,31} 4) la tecnología *Wireless Trusted*, la cual proporciona servicios de seguridad y privacidad en el manejo de la información.³²

De acuerdo con la información suministrada por Intel, estos procesadores estarán en capacidad de operar en un rango de frecuencias desde 312MHz hasta 624MHz y con una capacidad de memoria hasta de 64 MB.²⁶

El *core XScale* de Intel es una arquitectura tipo RISC de 32 bits basada en el *core ARM V5TE* con un subsistema de memoria tipo *harvard* formado por *caches* de instrucciones y de datos de 32KB con MMUs independientes y una *cache* de datos de 2KB que permite almacenar datos *streaming*. *Xscale* también posee una unidad de manejo de potencia, unidades JTAG, interfaces de *debugg*, interfaz de alta velocidad para la tecnología *Wireless MMX™*, y un coprocesador MAC que permite procesar eficientemente algoritmos de audio tipo multimedia. La microarquitectura tiene la capacidad de procesar datos tipo entero de 8, 16 y 32 bits en forma tradicional y datos tipo entero de 16 bits en modo SIMD.³³

Se debe destacar que las mejoras significativas en las capacidades de procesamiento de la familia PXA27x se deben principalmente a la integración de la tecnología *Wireless MMX™* y la tecnología *Quick Capture*, las cuales fueron mencionadas en forma breve anteriormente, y a continuación se detallará su descripción.

La unidad *Wireless MMX™* es un coprocesador acoplado a la microarquitectura *Xscale*, el cual integra funciones de tipo entero y multimedia en instrucciones SIMD de 64 bits, las cuales permiten procesar ocho datos en un solo ciclo. Adicionalmente, este módulo responde a modos de bajo consumo brindando la posibilidad de ser activado o desactivado de acuerdo con los requerimientos de la aplicación. La naturaleza programable de la tecnología *Wireless MMX™* y las unidades funcionales entre las que se encuentra un gran banco de registros,²⁷ permiten la implementación del estándar de codificación y decodificación de video MPEG-4 mediante un conjunto de instrucciones que proporcionan aceleración a aquellas funciones que resultan más intensivas desde el punto de vista computacional (estimación de movimiento, compensación de movimiento).

En módulos multimedia los procesadores de la familia PXA27x poseen un rico conjunto de funcionalidades. Para el caso de la captura de imágenes y video, los procesadores PXA27x introducen la interfaz configurable *Quick Capture* que proporciona una conexión directa del procesador con una amplia variedad de sensores CMOS. La interfaz recibe los datos y las señales de control del sensor y

proporciona el formato de representación apropiado para la transferencia de los datos a la memoria a través de subsistema DMA. La interfaz soporta diferentes formatos de datos como RAW, YCbCr 4:2:2, RGB 8:8:8 entre otros.³⁰ El módulo controlador LCD proporciona una interfaz entre los procesadores de la familia PXA27x y el módulo *display*. Específicamente el controlador soporta *display* tipo DSTN (Liquid Cristal Displays - Dual Scan Twisted Nematic (conocido como matriz pasiva)) y TFT (Liquid Cristal Displays - Thin Film Transistor (conocido como matriz activa.)) de una resolución máxima de 800 x 600 *pixels*. La configuración del controlador se hace a través de opciones programables para el tipo de display, resolución, *buffer frame* externo, formato del dato de salida entre otros parámetros de visualización. Internamente el controlador LCD está formado por un subsistema DMA dedicado de siete canales, un módulo de escalamiento, un módulo de conversión de espacio de color, tres RAM de 256x25 bits, y módulos de representación de datos para los diferentes tipos de *display*.

Para cumplir una de las principales restricciones que impone el diseño de aplicaciones móviles, la familia de procesadores PXA27x implementa la tecnología *Wireless SpeedStep*, la cual ofrece manejo dinámico de voltaje y frecuencia e implementa cinco modos de operación de bajo consumo: *Idle*, *Deep Idle*, *Standby*, *Sleep* y *Deep Sleep*^{29,34} que le permiten variar el consumo de acuerdo con el nivel de actividades monitoreado por la unidad de monitoreo de desempeño (PMU) y a la disponibilidad de los recursos.

**D. Procesador Dragonball
MC9328MX21 de Freescale**

Este procesador posee como CPU el *core* ARM926EJ-S de 32bits de la familia ARM9, integrado en un sistema on-chip (SoC) con módulos de *hardware* de aplicación específica que operan en forma independiente. La CPU ARM926EJ-S es un subsistema de alto desempeño, cuya velocidad de procesamiento puede llegar hasta 266 MHz con una frecuencia máxima de

reloj para el bus interno de 133 MHz. Posee una arquitectura *cache* tipo *harvard* y está formada por cuatro módulos principales: un *core* entero ARM9EJ-S, *caches* independientes de datos e instrucciones de 16KB y Unidad de Manejo de Memoria (MMU).³⁵ Adicionalmente la plataforma ARM integra el estándar AMBA-AHB^{18,19}, el cual permite realizar transferencias de datos entre módulos del sistema sin la intervención de la CPU en el flujo de datos.

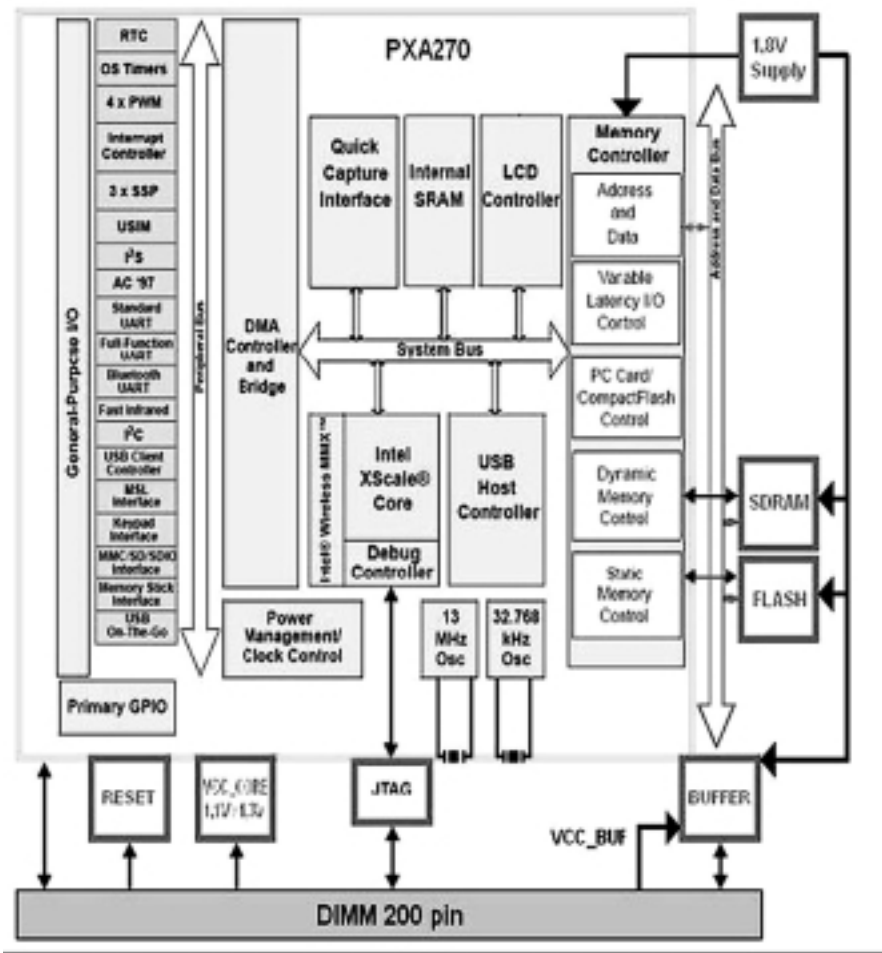


Figura 3. Arquitectura del procesador PXA270 de Intel.²⁷

En la Figura 4 se muestra el diagrama de bloques de los módulos que componen el procesador Dragonball. Entre los módulos específicos que posee este procesador cabe destacar la Interfaz de Sensor CMOS (CSI), el Acelerador Multimedia Mejorado (eMMA) y el Controlador de Display de Cristal Líquido (LCDC).

La Interfaz de Sensor CMOS (CSI) permite que el procesador i.MX21 se pueda conectar directamente a un sensor de imágenes CMOS externo. Este módulo soporta entradas de datos *YCC*, *YUV*, *Bayer*, o *RGB* de 8 bits^{35,36} y un tamaño de imagen máximo de 2560 x 1920. Internamente contiene una memoria FIFO de 32 x

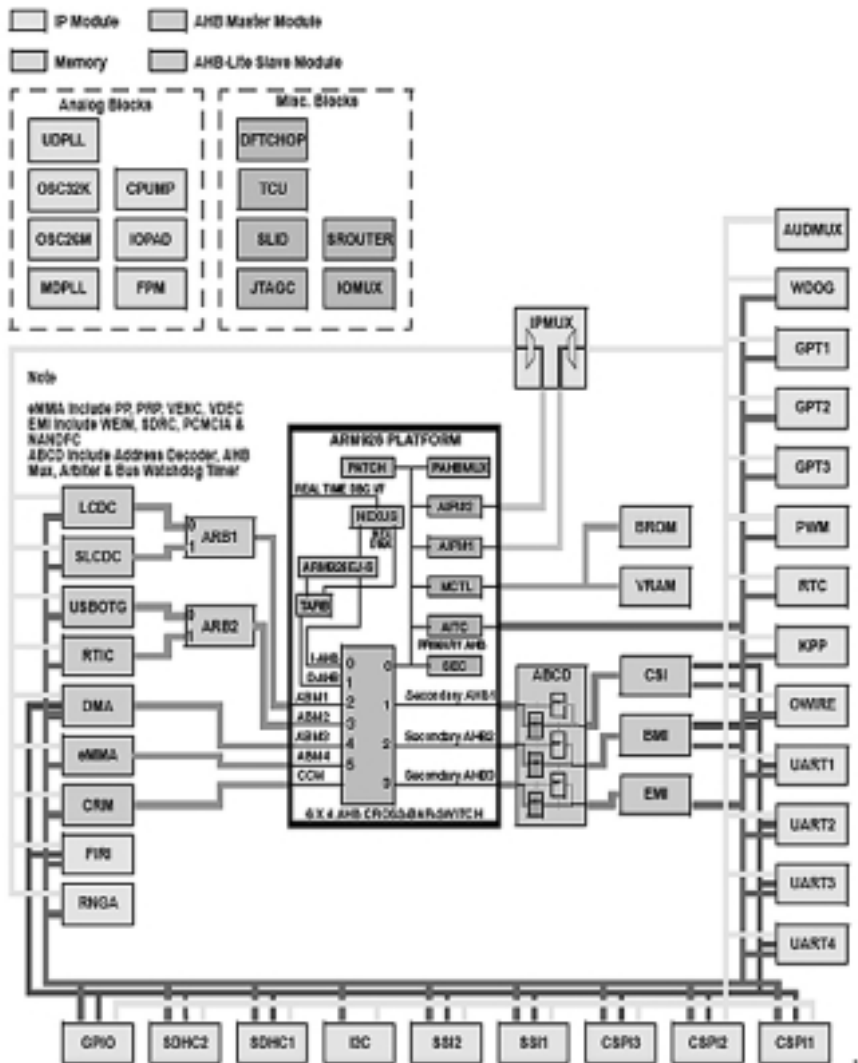


Figura 4. Espacio de memoria para el i.MX21.³⁵

32 palabras para el almacenamiento de píxeles de la imagen, una memoria FIFO de 16 x 32 palabras para la generación de datos estadísticos y una interfaz directa con el bloque de pre-procesamiento (PrP) del eMMA.

El Acelerador Multimedia Mejorado (eMMA) está formado por cuatro subsistemas: el pre-procesador, el post-procesador, el Codificador y Decodificador MPEG4 ISO/IEC 14496-2 de perfil simple con niveles de 0 a 3 de Hantro.³⁷ Estos subsistemas se conectan independientemente tanto al bus interno del procesador como al bus AMBA-AHB. El pre-procesador está formado por un *buffer* de entrada, un módulo de redimensionamiento y por un conversor de espacio del color; recibe *frames* desde 32 x 32 *pixels* hasta 2044 x 2044 *pixels* desde el sistema de memoria o a través de la interfaz directa que tiene con el módulo CSI y dispone de dos canales independientes en los cuales realiza el procesamiento y la distribución de los datos de 32 x 32, ya sea a un *display* o a la memoria. El post-procesador toma como entrada *frames* decodificados desde la memoria los almacena en el *buffer* Y (116x32) y el *buffer* UV (68x32) respectivamente y realiza sobre ellos el proceso de división en bloques, redimensionamiento y conversión de espacio del color para llevarlos finalmente desde un *buffer* de salida, a la memoria en el formato apropiado para la visualización.³⁵

El Controlador de Display de Cristal Líquido (LCD) permite al procesador mostrar datos en paneles LCD de color pasivo (CSTN) y de color activo (TFT) en una interfaz de 12, 16, y 18 bits y con una resolución máxima de 800 x 600.^{35,38} El flujo de datos en el

módulo LCD para el manejo del *display* tipo TFT, consiste en transferir los datos (píxeles de la imagen) desde el sistema de memoria por intermedio de un controlador DMA interno hasta los *buffers* de 32 x 32 de las interfaces *background* y *foreground*, respectivamente. Una vez los datos hayan sido procesados ingresan a un *buffer* intermedio, luego a la interfaz lógica y finalmente son enviados al *display*.

Uno de los aspectos arquitecturales más relevantes del procesador MC9328MX21 es el uso de AMBA-AHB en el esquema de comunicación, el cual es llamado específicamente para esta tecnología como *Multi-Layer AHB Crossbar Switch (MAX)*³⁵ Este esquema permite realizar transferencias de datos en forma simultánea y paralela entre diferentes módulos del sistema. El estándar se caracteriza por soportar múltiples maestros y ofrecer un alto desempeño y gran ancho de banda. El suiche MAX soporta seis maestros y cuatro esclavos, específicamente los módulos que son manejados como maestros son: *Icache* y *Dcache* que proporcionan rutas para acceder la memoria externa, el módulo DMA y periféricos de aplicación como el controlador LCD, la interfaz USB y el módulo eMMA.

Con el objetivo de reducir el consumo de potencia, el procesador MC9328MX21 soporta tres modos de operación (*run*, *standby*, *stop*) e incorpora además la implementación de la técnica *clock gating* dentro de los módulos para minimizar la energía consumida en el suicheo CMOS. Igualmente, para reducir el consumo de energía en estado *standby*, el procesador soporta la aplicación de técnicas de polarización y el manejo de

potencia dinámica (escalamiento de voltaje y frecuencia).

IV. ANÁLISIS DE LAS ARQUITECTURAS

En esta sección se realiza un análisis comparativo de las cuatro arquitecturas discutidas.

Un primer acercamiento a la evaluación cuantitativa en desempeño y consumo de potencia de algunos de los procesadores orientados a aplica-

ciones móviles y multimedia es aportado por *Synchromesh Computing* en su reporte técnico,³⁹ en el cual se comparan los procesadores i.MX21 de Freescale, PXA255 de Intel y el OMAP 1611 de Texas Instruments. Para la realización de las pruebas se dispusieron ocho benchmarks (Hint, Stream, MPEG4 Encode, MPEG4 Decode, Sisoft Sandra, BMQ, VObench, ByteMark) con el fin de medir diferentes parámetros en cada proce-

Tabla 2. Medidas comparativas entre procesadores multimedia (Los precios fueron establecidos mediante solicitud a distribuidores de estos procesadores)¹

	Nomadik STN8800 (sTMicroelectronics)	PXA270 (Intel)	Dragonball MC9328MX21 (Freescale)
Freq Procesador /Freq Bus	200 MHz	312MHz/104MHz 624MHz/208	MHz 266 MHz /133 MHz
Cache de Instrucciones	32KB	32KB	16K
Cache de Datos	16KB	32KB Y 2KB adicionales	16K
Memoria Interna RAM	Memoria SRAM on-chip de 48KB para almacenar los <i>frames</i> de video	SRAM de 256KB en bancos de 64KB	—
Potencia Modo Idle (mW)	100mW [22]	109 mW-260 mW (respectivamente a las frecuencias)	108mW
Potencia Modo Activo Solo manejo de UART (mW)	---	375 mW-925 mW (respectivamente a las frecuencias)	248mW
IC process	CMOS 130 nm	CMOS 130 nm	CMOS 130 nm
Precio en volúmenes de 10,000	\$15 USD ¹	a 312 MHz \$32 USD ¹	\$16.80 USD ¹

sador. Entre las conclusiones más importantes se pueden destacar:

- OMAP1611 presenta el menor consumo de potencia (debido a la baja frecuencia del *core* 180MHz), pero también presenta el peor desempeño para las aplicaciones MPEG4 debido probablemente al tamaño pequeño de la *cache* de datos (8KB) y a la baja velocidad del bus de memoria.
- PXA255 presenta el más alto consumo de potencia con un desempeño relativo alto.
- i.MX21 a pesar de su baja velocidad de reloj respecto al PXA255 obtiene un buen desempeño con un nivel de consumo medio. De acuerdo con los autores, este procesador presenta la mejor relación desempeño/milivatios.

Adicionalmente, el subsistema de *hardware CODEC MPEG4* fue evaluado solamente para el i.MX21 evidenciando en él, una gran mejora en el desempeño y una reducción en el consumo de potencia total del sistema, sobre todo para el caso de la decodificación.

En el caso de aplicaciones móviles la figura de mérito de la potencia se ha tornado tan importante como el desempeño. Generalmente las innovaciones arquitecturales han sido desarrolladas buscando un mejor desempeño sin considerar el impacto en el consumo de potencia.⁴⁰ Lo ideal es disponer de una estrategia de comparación de las arquitecturas, con base en el consumo de potencia por ejemplo basándose en *benchmarks*.

Debido a que no se dispone de medidas de potencia de las cuatro arquitecturas, se usan para este análisis

preliminar datos de publicaciones y reportes técnicos.

En la Tabla 2 se muestran los valores de los parámetros generales para cada uno de los procesadores. La arquitectura *MOBYDICK* no se especifica debido a que su implementación se reportó conectado varias FPGAs y procesadores y no como un circuito integrado único.

De la Tabla 2, se puede colegir, en relación con el sistema de memoria, que todos los procesadores poseen una arquitectura tipo *harvard*, con tamaños de *cache* entre 16KB y 32KB con diferentes grados de asociatividad. Esta estructura de almacenamiento responde a las recomendaciones presentadas en⁴¹ que establece para aplicaciones multimedia, *caches* de datos con tamaños entre 16KB y 32KB debido a que ofrecen una baja tasa de error en los accesos a memoria (*cache misses*) comparada con otras aplicaciones de uso corriente. Igualmente para la *cache* de instrucciones se recomienda un tamaño de 32KB o mayor para reducir la tasa errores de acceso originadas por aplicaciones multimedia más complejas. Por otra parte los diferentes sistemas de memoria de estos procesadores están enriquecidos con técnicas orientadas a mejorar el desempeño y el ancho de banda tanto de las memorias *cache* como de la memoria principal, entre ellas se pueden citar: el uso de técnicas como *write buffer*, *write-through* y *write-back* entre otras.

Respecto a la organización arquitectural de las plataformas, se puede observar que como resultado de las mejoras en los procesos tecnológicos, cada vez se integran mayor número

de componentes en un solo chip (*System-on-Chip* SoC) y la comunicación entre estos suele ser más crítica. Frente a esta situación la tendencia en el diseño de arquitecturas de comunicación está cambiando del tradicional sistema de bus compartido a estructuras de comunicación más complejas, basadas en múltiples buses que incrementan el ancho de banda.⁴² Los procesadores STN8800 e IMX21 implementan el estándar AMBA, mientras que el procesador PXA270 posee una configuración tradicional en el sentido que tiene un bus central de comunicación que conecta módulos de *hardware* de aplicación específica a la CPU. De acuerdo con los resultados reportados por⁴³ un bus compartido de 32 bits proporciona menor desempeño que una configuración de buses múltiples (3 buses) de igual tamaño. Adicionalmente, cada vez más se evidencia que la arquitectura de comunicación elegida no solo afecta el desempeño sino también el consumo de potencia. De acuerdo con⁴⁴ para un esquema AMBA de dos AHB-Master, dos AHB-slaves, un puente AHB-APB^{18,19} y tres periféricos APB el mayor consumo de potencia es presentado por los componentes lógicos del sistema, también se demuestra que el tipo de transferencia, el uso de *pipeline* y la complejidad del sistema que aumenta con el número de módulos al conectar influyen significativamente en el consumo de potencia. Actualmente el área de diseño de arquitectura de comunicación *on-chip* se considera un campo abierto para experimentar. Muestra de ello son aquellas publicaciones que proponen otros esquemas como en⁴² donde se habla del uso de redes directas, indirectas e híbridas, en⁴⁵

que se propone el uso de microrredes reconfigurables. Es de anotar que otra tendencia es el uso de sistemas asíncronos⁴⁶ que permiten usar las técnicas de suicheo de voltaje y escalamiento de frecuencia para reducir el consumo de energía.

Respecto al consumo de potencia, y a los datos mostrados en la Tabla 2 para esta variable, es importante anotar que existen grandes dificultades en cuanto a los métodos de medida ya que según el Consorcio de *Benchmarks* para Microprocesadores Embebidos (EEMBC)^{47,39} en la actualidad no se han definido metodologías bien establecidas para estimar el consumo de energía considerando aplicaciones móviles de procesamiento multimedia. Cada fabricante ofrece una especificación de consumo de energía típico que es difícil de comparar con otros, sobre todo con fuentes de información tan limitadas como la del procesador STN8800.

Finalmente vale la pena anotar que con respecto a la tecnología de diseño, los procesadores Dragonball y STN8800, presentan un sistema abierto en el sentido que integra tanto tecnología propietaria en algunos módulos de *hardware* como subsistemas IP y estándares abiertos (e.g estándar AMBA), mientras que la tecnología del procesador Intel es totalmente propietaria, lo que podría ocasionar limitaciones para que los diseñadores desarrollen aplicaciones.

CONCLUSIONES

En las arquitecturas embebidas orientadas a aplicaciones multimedia móviles, el sistema de memoria y la estructura de comunicación tienen un

impacto considerable en el desempeño, el costo y el consumo de potencia del sistema. La optimización de estos tres parámetros simultáneamente en el proceso de diseño, es incompatible, ya que típicamente la mejora de uno de ellos implica el detrimento del otro. La clave para el diseño de estas arquitecturas consiste en encontrar un punto óptimo para la relación costo/potencia considerando un desempeño suficiente para llevar a cabo la aplicación deseada de forma conveniente.

La implementación de las diferentes técnicas de reducción de consumo de potencia (estática y dinámica) en todos los niveles del diseño de las arquitecturas mostradas, evidencia la gran importancia de maximizar los recursos de la arquitectura con el fin de incrementar el tiempo de operación de las fuentes de suministro de energía, ya que los nuevos dispositivos requieren operar en un ambiente de movilidad con baterías de duración limitada.

Como trabajo futuro se buscará establecer metodologías que permitan medir el consumo de energía en arquitecturas embebidas orientadas a aplicaciones multimedia móviles, realizando pruebas reales con un conjunto de aplicaciones con los procesadores Dragonball MC9328MX21 de Freescale y PXA270 de Intel, con el fin de establecer medidas comparativas confiables que soporten criterios de selección frente al consumo de potencia, el costo y el desempeño.

AGRADECIMIENTOS

Los autores agradecemos a la Universidad de Antioquia y a Colciencias por el apoyo brindado para la realización de este trabajo.

BIBLIOGRAFÍA

1. Nuwan Jayasena, Suzy Rivoire, "Advanced Topics in Computer Architecture Chip Multiprocessors and Polymorphic Processors". Stanford University, April 3rd, 2003. <http://www.stanford.edu/class/ee392c/notes/lec07/notes07.pdf>
- 2.A. Shahbahrami, B. Juurlink Stamatias Vassiliadis, "A Comparison between Processor Architectures for Multimedia Applications". http://ce.et.tudelft.nl/publicaties/933_509_shahbahrami.pdf
3. J. Fritts. "Architecture and Compiler Design Issues in Programmable Media Processors". PhD thesis, University of Princeton, 2000.
4. D. Talla, Lizy K. John, V. Lapinskii, and Bran L. Evans, "Evaluating Signal Processing and Multimedia Applications on SIMD, VLIW and Superscalar Architectures". Department of Electrical and Computer Engineering, The University of Texas, Austin, TX 78712.
5. D. Talla., "Architectural Techniques to Accelerate Multimedia Applications on General Purpose Processors". PhD thesis, University of Texas at Austin, 2001.
6. Telos EDV Systementwicklung GmbH. <http://www.telos.de/trimedia/>. Fecha de consulta: junio de 2005.
7. <http://www.samsung.com/Products/Semiconductor/SystemLSI/MobileSolutions/Mobile>
ASSP/Mobile Computing/
S3C24A0/S3C24A0.htm

8. Freescale. <http://www.freescale.com>. Fecha de consulta: junio de 2005.
9. Intel Corporation. <http://www.intel.com>. Fecha de consulta: junio de 2005.
10. Christoforos E. Kozyrakis and David A. Patterson., "A New Direction for Computer Architecture Research". Computer Science Division. University of California at Berkeley, CA 94720. <http://csl.stanford.edu/~christos/publications/1998.IEEEComputer.Direction.pdf>
11. Christoforos Kozyrakis and David Patterson, "Vector Vs. Superscalar and VLIW Architectures for Embedded Multimedia Benchmarks". In the Proceedings of the 35th International Symposium on Microarchitecture, Instabul, Turkey, November 2002.
12. Rixner, W. Dally, U. Kapasi, B. Khailany, A. Lopez-Lagunas, P. Mattson, J. Owens, "A Bandwidth-Efficient Architecture for Media". Processing, IEEE/ACM International Symposium on Microarchitecture (MICRO), pp. 3-13, Dallas, TX, November 1998.
13. J. Fritts, W. Wolf, and B. Liu. "Understanding Multimedia Application Characteristics for Designing Programmable Media Processors". In Proc. SPIE Photonics West Media Processors, pages 2-13, Jan 1999.
14. K. Diefendorff and P. K. Dubey. "How Multimedia Workloads Will Change Processor Design". In IEEE Computer, pages 43- 45, September 1997.
15. Paul J.M. Havinga , "Mobile Multimedia Systems", Ph.D. thesis University of Twente, February 2000, ISBN 90-365-1406-1.
16. Xilinx Company. <http://www.xilinx.com>. Fecha de consulta: junio de 2005.
17. STn8810/STn8811/STn8812: Trio of Nomadik application processors bring multimedia to next-generation mobile devices. February 2005 STMicroelectronics.
18. TA305 Technical Article "Nomadik - Open multimedia platform for next generation mobile devices" 2004 STMicroelectronics.
19. Sudeep Pasricha, Nikil Dutt, Mohamed Ben-Romdhane, "Automated Throughput-driven Synthesis of Bus-based Communication Architectures". 2005 IEEE. ASP-DAC 2005. http://www.cecs.uciedu/conference_proceedings/asaspdac_2005/pasricha_automated.pdf
20. AMBA Specification (Rev 2.0). © Copyright ARM Limited 1999. <http://www.arm.com>
21. STMicroelectronics. <http://www.st.com>. Fecha de consulta: junio de 2005.
22. Max Baron, "Nomadic Leads Pack Winning Chip Combines Legacy Silicon and Software With New Architecture". In Microprocessor Report February 9, 2004.
23. Product Brief, Intel® PXA255 Processor with Intel XScale Technology. 2003 Intel Corporation.
24. Datasheet, Intel® PXA255 Processor Electrical, Mechanical, and

- Thermal Specification. February, 2004 Order Number: 278805-002.
25. Intel® PXA26x Processor Family: Electrical, Mechanical, and Thermal Specification. April 2003. Order Number: 278640-002.
 26. Product Brief, Intel® PXA27x Processor Family. 2004 Intel Corporation. Order Number: 253820-002.
 27. N.C. Paver, M.H. Khan, B.C. Aldrich, "Accelerating Mobile Multimedia with the Intel® PXA27x Processor Family". Intel Corporation, 1501 S. Mopac, Suite 400, Austin, Texas, USA, 78746.
 28. Wireless Intel SpeedStep® Power Optimizing power consumption for the PXA27x processor family. White Paper. Intel®Manager:Corporation 2004. Order Number: 300577-001.
 29. Intel® PXA27x Processor Family Power Requirements Application Note. Intel Corporation, 2004 Order Number: 280005-002.
 30. White Paper, Intel® Quick Capture Technology for the Intel® PXA27x Processor Family. 2004 Intel Corporation. Order Number: 300873-001.
 31. White Paper, Optimization Technology for the Intel® PXA27x Processor Family (Performance and Power Savings for Wireless System and Application Development). 2004 Intel Corporation. Order Number: 300869-001.
 32. Intel® Wireless Trusted Platform: Security for Mbile Devices. White Paper, Intel Corporation 2004. Order Number: 300868-001.
 33. Intel® XScale Microarchitecture. Technical Summary. Intel Corporation, 2000.
 34. Matt Gillespie, "Power Management in the Intel® PXA27x Series Application Processors". April 2005.http://cache-www.intel.com/cd/00/00/21/70/217033_217033.pdf
 35. i.MX21 Applications Processor. Reference Manual. C9328MX21RM/D Rev. 1.1, 11/2004 © Freescale Semiconductor, Inc. 2004.
 36. Application Note AN2676/D. Rev0 , 03/2004, Image Capture with i. MX21 Motorola, Inc., 2004. [http:// www.freescale.com](http://www.freescale.com)
 37. Clint Powell, White Paper. i.MX Applications Processors with Hantro's Multimedia Framework. Freescale Semiconductor Inc. Marko Nurro, Hantro Products Oy IMXHANTROWP/D Rev. 2, 09/2004.
 38. Amanda McGregor, "Using the MC9328MX21 (i.MX21) to Create a VGA Digital Photo Album" Application Note AN2868 Rev. 1, 11/2004. Freescale Semiconductor, Inc., 2004.
 39. "The Freescale Semiconductor I.MX21 Processor". Synchromesh Computing. 2004.
 40. J. Hennessy and D. Patterson, "Computer Architecture a Quantitative Approach". Morgan Kaufmann Publishers 2003.
 41. S. Sohoni, Z. Xu, R. Min, and Y. Hu, "A Study of Memory System Performance of Multimedia Applications," In Proceedings of the ACM Joint International

- Conference on Measurement & Modeling of Computer Systems (SIGMETRICS 2001), (Cambridge, Massachusetts), pp. 206-215, June 16- 21 2001.
42. Zheng Yuan and TalaI Arnaout, "Architectural Level Power Consumption of Network on Chip". Seminar: Network on chip. 2003.
 43. K.Lahiri, A.Raghunathan, S.Dey, "Performance Analysis of Systems with Multi-Channel Communication Architectures", in Proc. Intl. Conf. on VLSI Design, pp.530-537, Calcutta, India, January 2000.
 44. Kanishka Lahiri and Anand Raghunathan , "Power Analysis of System Level OnChip Communication Architectures". NEC Laboratories America, Princeton, NJ. 2004 ACM.
 45. Luca Benini and Giovanni De Micheli, "Networks on chips: A new SoC paradigm". IEEE Computer. Enero 2002.
 46. MPF hosts premiere of ARM1136: cores combine ARMv6, SIMD, eight-stage pipeline, and more.(Microprocessor Forum 2002) Microprocessor Report; October 01, 2002; Levy, Markus.
 47. Embedded Microprocessor Benchmark Consortium. <http://www.eembc.org>. Fecha de consulta: Julio de 2005.

CURRÍCULOS

Mónica Aydé Vallejo Velásquez.

Recibió el título de ingeniera electrónica por la Universidad de Antioquia - Colombia, en el 2003 y actualmente se encuentra desarrollando su programa de maestría en ingeniería electrónica en la Facultad de Ingeniería desde el 2004. Sus áreas de interés son la computación móvil, el diseño de hardware basado en FPGAs y las comunicaciones inalámbricas. Trabaja con el Grupo de Investigación en Microelectrónica y Control de la Universidad de Antioquia.

José Edinson Aedo Cobo. Obtuvo el título de ingeniero electricista por la Universidad del Valle en 1987 y el grado de maestría y de doctorado por la Escuela Politécnica de la Universidad de Sao Paulo-Brasil en 1992 y 2000, respectivamente. Actualmente es profesor del Departamento de Ingeniería Electrónica de la Universidad de Antioquia - Colombia y dirige el grupo de investigación de Microelectrónica y Control en dicha Universidad. Sus áreas de interés cubren el diseño de circuitos integrados de dedicación específica, arquitecturas de hardware para computación móvil e inteligencia computacional. ☼